Morvyn N. Nyakudya (Botswana)

Scanning the Botswana computing environment for knowledge of open source software

Abstract

Thousands of extremely useful open source applications and utilities are available free under different licenses. These range from, among others, anti-spyware, audio, operating systems, educational software, business applications and programming utilities. After many studies on the subject concerning open source software (OSS), several authors have bluntly concluded that OSS is actually better than its proprietary counterpart. This assertion may not be true for developing countries such as Botswana. A necessary precondition for OSS adoption is knowledge for open source. To this effect, this article analyses the extent to which companies in Botswana have embraced the OSS initiative as alternative to proprietary software. The question is: "Do they have enough knowledge of OSS products availability?" Data were collected from 62 small organizations that engage in some form of information technology (IT) in their daily operations and analyzed using simple descriptive statistics such as mean and standard deviation. The findings indicate some three important aspects: (1) all organizations employ many forms of IT to a great extent coupled with a negligible OSS component; (2) organizations have invested a lot of capital into their current IT systems and therefore regard OSS as an inferior and ineffective alternative compared to the licensed software that they use; (3) although IT professionals have abundant knowledge of OSS products, they perceive that the power to switch from proprietary to OSS solely rested with the company owners and their top management.

The major implications of the findings for future research have been identified and presented.

Keywords: Botswana, OSS, IT, proprietary software.
JEL Classification: O31, O33.

Introduction

Rapid technological changes are increasingly affecting the terrain of the information and communications technology (ICT) sector around the world. These rapid transformations require a firm grip for organizations to move along with the changing times for them to sustain a competitive advantage. The increasing intensity of competition has resulted in some organizations changing their mode of ICTs, opting for open source software (OSS) packages that are freely available and flexible. The advent of OSS has aided those firms with limited IT budgets, (Waters, 2007) to enjoy the benefits offered by the free software. These benefits include, among others, freedom of use, power (speed and scale), networking (OSS is more network friendly), customization, open formats provision as well as being a software of choice for schools. Laden with all these advantages, OSS can be a necessary and essential convenient alternative to proprietary software which remains closed and very costly.

The purpose of this paper is therefore to investigate whether, in Botswana, firms have any knowledge of OSS taking into consideration that knowledge is a pre-requisite for final adoption. The paper argues that if these computing firms have a well-grounded knowledge and understanding of the various OSS products, then they can consider innovating by increasingly utilizing the products whose development community is constantly motivated to make the same products even much better and more superior than their proprietary counterparts. The study in this paper is based on companies listed in the Botswana Confederation of Commerce and Industry Manpower (BOCCIM) 2009 directory of companies. The companies are small and medium enterprises (SMEs) and were randomly selected (each company has an equal chance of being selected) to create a sample of 62 subjects.

1. OSS: an alternative platform

The world computing arena is witnessing a major shift from a traditional, proprietary platform to an OSS evolution. In developed nations, some managers have opted for OSS, partly or wholly, while others have opted for a hybrid between OSS and proprietary. Yet others have chosen to make the OSS run in the background while proprietary remains the visible component. The OSS phenomenon advocates that software is free, readily available and flexible. On the other hand, the traditional software development platform requires that software must be fully developed and tested before anyone can have access to it. The purpose of this article is to gauge the extent to which programmers and other IT professionals in Botswana know about the different OSS products that are readily available for free on the Internet. It is essential that the open source initiative (OSI) is described in detail. In this regard, we detail the description of OSS in terms of the source code availability, its advantages and disadvantages, licensing issues, community activities, and obligations of the communities, organizational structures and security, among others.

OSS is subjected to scrutiny with bugs being found and removed quickly; unlike in proprietary software where problems are discovered late, sometimes resulting is very damaging consequences. Because of the strict peer-review process in open source, the resulting code base is more reliable than the closed software code which is not known by anyone except the vendor. Thus, the concept of open source is that programmers have the ability to read, modify and redistribute code resulting, eventually, in fine tuned superior software packages. In other words, through the involvement of many people, OSS evolution results in good and reliable software. The involvement of users will mean that they can innovate easily, suggesting, in the process, brilliant ideas to be incorporated for the purpose of meeting their needs. Users can be part of the developing process or they can engage a group of developers (for example, LibLime) for this task. The result of innovation means quick turnaround time on projects as compared to closed software where only the vendor has the ability to change software. Many authors have reported on this issue of OSS reliability as described in the following sections.

According to Lattermann and Stieglitz (2005), del Amo (2007), Bessen (2005), Johnson (2006) and Tere (2006), OSS is software which can be used, studied, modified and then redistributed with no restrictions. Sowe et al. (2007), Demaziere et al. (2007), Nieuwenhof (2008) and Riehle (2007) have defined OSS in light of zero acquisition costs and licenses for free distribution of software while Coar (2006) coined the OSS definition in terms of ten features that described, among others, free distribution, source code, derived works, and integrity of the author's source code, non-discrimination and licensing issues. Lattermann and Stieglitz (2005) and Millar, Choi, Russell and Kim (2003), cite Linux, Apache, Perl, PHP and MySQL in their definitions of OSS. Muir (2005) suggests that the OSS library community should urge people to experiment with software packages or even create their own.

In more definitions, other authors have wondered why communities work with no contracts but whole heartedly and professionally producing public goods of high value, Demaziere, Horn and Zune (2007), Freeman (2007), Millan, Choi, Russell and Kim (2005) and Alexy et al. (2009). They question how these community members are committed to put up a public good to use when they cannot engage in face-to-face talk and with no remuneration: mundane but necessary activities, Bessen (2004).

In traditional software, vendors must develop, deploy and improve their software for the user community, catering, in the process, for the wide diversity that exists in the market place. On the contrary, open source software is developed and improved by the users in the communities – the development is driven by the users of the systems. The user community makes decisions as to what features to include in order to make the software look good. To further highlight these concepts of "openness" and "closeness", Nieuwenhof (2008) compared OSS with proprietary software citing the closed nature of proprietary software resulting in the incapacity of programmers to make any modifications. For computer programmers to produce good software of quality, he argues, on another matter, traditional management advocate for proper organizational structures that deal with compensation among many other issues. Freeman (2007) cites intrinsic and extrinsic motivation as the main drivers. Henkel (2008) discusses the principal-agent issue, noting that this problem is a result of double loyalty of those in the OSS community. These and other authors have loaded literature with exciting OSS initiatives going on around the world.

The list of OSS utilities continues to grow as programmers wrestle to write the "best" software that can outperform the traditional proprietary software. These utilities cover many essential areas of today's modern business. The finance module, for instance, covers such areas as enterprise resource planning (ERP) solutions, Java-based quantitative finance framework and institution management software. Other areas of importance in business solutions include, among others, mathematics, computer simulation, science, geographical information systems (GIS), bioinformatics, statistics, artificial intelligence (computer vision, robotics, planning, machine learning, etc.), networking/Internet, educational, theology and games. Healthcare, media, operating systems, programming languages, security and many more software utilities have been written and remain not taken up or are unknown by many computing houses.

The question is whether Botswana has knowledge of this vast amount of work that has been done towards the OSS cause. If the answer to this question is largely "yes", then we argue that Botswana can be ready to adopt OSS as a software platform of choice.

This paper seeks to scan the Botswana computing environment for OSS knowledge. It is imperative however that we first describe some important open source software features. These features revolve around licensing issues, community activities, obligations as well as security and adoption issues.

1.1. OSS features. OSS is characterized by common features, the most important of which are the OSS licenses, OSS communities, OSS obligations and OSS security.

1.1.2. OSS licenses. Any software that runs does so under some licensing terms. OSS is vulnerable to being locked by malicious programmers and so licenses have kept these expensive public goods "unlocked" all the time. Nieuwenhof (2008) and del Amo (2007) note that the Forum for Software Foundation (FSF) and Open Software Initiative (OSI) are the two bodies that created the licenses for OSS. Under the licensing agreement, source code can be read, copied, adapted and redistributed using the *copyleft* method: the *copyleft* unlocks the copyright. The *copyleft* promotes free distribution of software and prevents any software under the General Purpose License (GPL) from becoming proprietary providing total liberty in using the code with no one monopolizing on the gesture extended by the license (Demaziere, Horn and Zune, 2007; Riehle 2007; and Lattermann and Stieglitz, 2005). Comparing commercially produced software and OSS, Lattermann et al. (2005) discuss the rights around permission to replicate software, free availability of source code and the propagation of software licenses for the benefit of all. Nieuwenhof (2008) asserts that licensing schemes are needed to enable anyone to use, read and modify the source code of computer programs.

The issue of technology transfer to developing nations is gaining momentum although the cost of paying for proprietary software is a key hindrance to the process. However, Alkhatib, Anis and Noori (2008) stress that users in the open source forum often get their product at no or minimum cost – this is encouraging for developing nations. Only special arrangements that warrant payment for developments and/or customization may be entered into between the developers and the users.

Many a time, organizations or individuals intending to utilize open source software often try to ignore the rigors of going through the licensing of the products – a waste of time or just lack of interest. It must be noted that OSS licensing is not restrictive process (compared to proprietary software licensing) as users tend to think and so it is advisable that any would-be open source software user manages this issue seriously. In this line of argument, McGhee (2007) discusses a host of unique legal licenses associated with OSS and warns that they must be evaluated before any gains can be realized, concluding with the observation that OSS can help companies realize tremendous gains if they use it intentionally or a complete nightmare logistically, economically and legally if used otherwise. In the same vein, McGhee (2007) concludes by warning users against ignoring OSS compliance.

For Botswana, the issue of licensing equally applies although most of their licenses are highly restrictive proprietary software related.

1.1.3. OSS communities, activities and motivation. "Why do programmers engage in necessary but mundane tasks of programming complex software?" Bessen's (2004) concern seems to ask. In the literature communities are heavily engaged, motivated but have no contractual obligations and fall under no organizational structures what so ever. Demaziere, Horn and Zune (2007) and Freeman (2007) describe free OSS communities as made up of individuals who work together in harmony, are orderly and share common social rules in order for them to function effectively. They regard software to be a public property which must be available to all, at no cost, with total freedom to use, read, modify and redistribute. Intrinsic motivation, relating to a sense of creativity and extrinsic motivation associated with expected future returns as well as personal needs for software, are extensively discussed and explained by Henkel (2008), Lattermann and Stieglitz (2005), Riehle (2007), Fitzerald (2004), Krogh et al. (2003), Krishnamurthy (2006) and Learner et al. (2001), among others.

The principal-agent issue and intellectual property (IP) (Henkel, 2008; McGowan et al., 2007), reputation building and volunteering – no compensation (Riehle, 2007), career concerns (Lattermann and Stieglitz, 2005), and tangible products creation (Sowe, Angelis, Stamelos and Manolopoulos, 2007) are some of the characteristics of OSS communities. Bagozzi and Dholakia (2006) made a study of participation in Linux user groups (LUG) noting the effects of attitudes, perceived behavioral control, identification with open source movement, positive and negative anticipated emotions as what measure the participation of communities. A software development group, LibLime, also promotes collaboration by passing projects to user groups within communities.

2. Obligations

Different groups in each community have some obligations that they have to fulfill.

Agerfalk and Fitzgerald (2008) present the obligations from the perspective of the customers, project managers and those of the communities.

2.1. Customer obligations. Customers must ensure that they put forward clear and explained requirement specifications for the system developers (programmers) to provide the services. They are expected to pay for their customization on time. Agerfalk and Fitzgerald (2008) propose that there should be close project monitoring as well as meetings attendance, adding that the holding of regular discussions should be a must. They cite examples of such conferences as the Apache Conference in the USA and the GNOME annual project conferences, to cite just two. The customer is also encouraged to own the project so as to promote growth and sustenance.

2.2. Community obligations. Agerfalk and Fitzgerald (2008) propose clear authority structures with decision-making rights, reporting structures in projects, roles and responsibilities. On solving a problem or completing a job, the community should be in charge with minimal customer involvement. The turnover in the community must be low: the community should work together as a team for long periods. The customer is expected to benefit much from the community work, transferring skills, knowledge and expertise.

2.3. Project managers. The top management priority is to champion the project to convince developers why the OSS strategy is a sensible venture to embark on. They should also be able to appoint key staff such as the OSS Program Director who engages with the community ensuring that quality projects are delivered quickly (Agerfalk and Fitzgerald, 2008).

3. OSS security

A concern of any IT system is its security. Among all challenges, perhaps the security issue is the most intriguing of all. An information technology (IT) infrastructure is protected by complex security software and other hardware gadgets and organizations allocate huge budgets towards this. Security of software is paramount. For "closed" software, it must be noted that, the support or future development depends on the owner or the developer only. In the case that the vendor busts, the product support also ceases, putting an organization into trouble. On the other hand, open source software is supported by many providers. If, say, the original creators of OSS go under, there are other groups, with different motivations and attitudes, to take over the development and support of the product. Various proponents of OSS have put up a strong argument for OSS security, arguing that the security of OSS matches, or even surpasses, that of proprietary software (Mohamed, 2008). Mohamed (2008) further argues that there are so many people who are working with OSS source code to the extent that the discovery of potential risks is quick and the fixing of the bugs, equally so. This is done quickly using the mailing lists, dispatching and exchanging patches over the Internet, testing the patches (peer reviewed) and releasing new systems. It is also encouraging to note that RedHat and Ingress are building advanced security systems in the form of fine-grained access control and security auditing and encryption.

In the same vein, Cox, Runge and van de Ven (2005) discuss security features in a RedHat Enterprise Linux which makes it the industry-leading commercial Linux solution: *Execshield* and *PIE* which offer great resilience to attacks, *SELinuse* for access control, *Compiler* and *library* enhancement among others.

3.1. To adopt or not to adopt. Organizations that are planning to adopt go through extensive and sometimes rigorous planning activities in order to assess, among others, the switching costs, security, external support and technical expertise of their computer departments. In this context, Ven, Verelst and Mannaert (2008) extensively discuss why firms should adopt or not, producing an options matrix showing claims and counterclaims based on five fundamentals: cost advantage, source code, software maturity, vendor lock-in and external support. Table 1, borrowed from Ven et al. (2008), presents the findings.

Table 1. Claims and counterclaims about open source software

Factor	Claims	Counterclaims
Cost advantage	OSS is free of charge. Linux can lower hardware costs.	Enterprise Linux isn't free of charge. Dual licensing might require a commercial license. Unclear total cost of ownership. Switching costs can be high.
Source code	Source code availability leads to higher quality, enables customizations, provides more choice and control, and provides more trust in the software. Source code can be important when developing products based on OSS because it provides more insight.	Lack of knowledge to apply modifications. Lack of need to apply bug fixes. Source code might not matter to organizations.
Maturity	Linux OSS is reliable. Category killers in horizontal domains.	Linux OSS is unreliable.
Vendor lock-in	OSS avoids vendor lock-in.	Choice for enterprise Linux might be mandated by external vendor. Still dependent on OSS vendor for updates, services and support.
External support	External support is important for OSS. Support for OSS is available from commercial vendors.	Type of required support differs. Support is lacking for some types of OSS.

Source: Should you adopt Open Source Software? By Ven, Verelst and Mannaert (2008, pp. 55).

They close their discussion by suggesting that decision-makers must consider organizational specifics before a final decision is taken to adopt. Through this scanning of organizational specifics, organizations are able to make an undertaking whether to adopt and if so, when and how. Tong (2004), on cost, agreed with Ven et al. (2008), asserting that initial acquisition costs are negligible and download of software such as Apache is free, making the initial OSS acquisition costs much lower than the cost of buying proprietary software. Henkel (2008) cites Nokia and Philips as having widely adopted some OSS as well as the OSS development approach. But he also proposes a hybrid strategy which allows firms to consider a position midway between purely open (OSS) and purely proprietary. Larsen, Holck and Pedersen (2000) focus their argument for OSS adoption on decision-making. They argue that larger firms are skeptical about OSS replacing proprietary software completely. Further, they assert that OSS in large organizations is run together with proprietary software packages opting to keep OSS invisible. Full adoption only comes after proper planning is put in place. Crisis adoption is however possible as noted by Vaisam (2007) who gives an account of how Iran has largely adopted OSS due to the U.S. embargo on its economy. Iranian universities produce software engineers, most of who are women, and are constantly developing software for the nation using the OSI model.

Although literature has been lauded with glossy advantages of OSS, the software nevertheless has its own drawbacks. Perhaps, it is now proper to have a look at these disadvantages. First, a project, if not supported, can die. In other words, there is no guarantee that projects will be carried out since the developers have no obligations, no remuneration, no organizational structures and no motivation (Bessen, 2004; Demaziere et al., 2007; Freeman, 2007; Henkel, 2008; Lattermann et al., 2005; Riehle, 2007; Sowe et al., 2007; and Bagozzi et al., 2006). Developers may just find it interesting to work on a project but if this interest fades or no longer exists, the project will not be completed.

Again, the IP rights issue is developing into an important phenomenon regarding open source projects worldwide. Some countries are now patenting software and so communities will not be able to experience one of the tenets of the OSI – freedom, because of the IT rights clause. Countries will claim violation of the IP rights for patented products. Further, not much publicity is given about the existing projects as well as their current status and hence can die silently.

4. Methodology

Fourteen different OSS software groupings were defined (the fourteen groups identified are the most common of the OSS family). Respondents were asked to indicate the extent of their knowledge of each product, based on a five-point Likert scale (1 = no knowledge, 2 = less knowledge, 3 = average knowledge, 4 = more knowledge, 5 = abundant knowledge). This scaling format is commonly used for assessment of attitudes towards subject areas. Follow-up telephone calls and email reminders aided in boosting the questionnaire return rate to a great extent: each respondent was sent a reminder at least twice. Initially, the researcher's intention was to target purely computing companies only but due to pre-test non-committal behavior of some of the respondents, other sectors were then considered. As earlier mentioned, the sampling frame in this paper is made up of companies registered with the Botswana Confederation of Commerce and Industry Manpower (BOCCIM) and falling within the manufacturing (14), retail (10), agriculture (2), computing (16), service (10) and education (16) sectors. The aim of constructing a mixed sample is to ensure that adequate comparison is carried out across different sectors. Out of the 75 questionnaires sent out, 65 (87%) were returned and, of these, 3 (1%) were found to be spoilt due to missing data. A total of 62 (99%) of those returned were found usable. The questionnaire was divided into two parts. Part 1 dealt with the nature of business of the organization selected as well as the position of the respondent in the company. Part 2 covered questions to address the levels of knowledge each respondent had on each of the OSS products identified for the purpose of this study. These, as earlier mentioned, included, among others, security, programming, file management, games and operating systems, Table 2 and were used to measure the knowledge construct. Respondents' levels of knowledge are treated as a matter of perception rather than of fact. Therefore, the best statistical method for this measurement is descriptive statistics focusing mostly on mean and standard deviation. Some general questions were also included in part 2 where respondents were expected to explain or list some products under their preferred categories.

Each variable of interest started with the phrase "*knowledge of*", for example, "*knowledge of OSS Operating Systems*". All the respondents worked as either IT professionals or were high-end IT users who had a good understanding of software issues. The questionnaire was pretested for validity at three top IT companies in Gaborone and once it passed this test, was dispatched to the respondents. Knowledge of OSS products is very important and comes before testing the readiness of the same products. Moon et al. cite knowledge and adoption at national level using policy mechanisms/regulatory approaches. They argue that by 2001, Peru, Brazil, Argentina, France and Mexico could use OSS on government computers; Germany, Spain, Italy and Vietnam, among others, followed suit. Camino et al. (2005) also analyze the impact on social welfare of government policies supporting OSS. They divide mass-market consumers into two groups: those who have the knowledge about the existence of OSS and those who are not informed. Further, they argue that government can play a pivotal role they term government intervention which can come in three forms: (1) mandated adoption (in schools, public agencies, and universities); (2) information provision (where government takes a role to inform users about OSS; and (3) subsidy (in which government makes a payment to those who adopt OSS).

5. Findings

The purpose of this study was to investigate Botswana's awareness of OSS. The basis for reporting in this study is whether IT professionals in the companies investigated had some knowledge (working or otherwise) of OSS products. To confirm that general OSS knowledge is a precondition for this research, respondents were asked whether they knew anything about OSS. All participants (100%) responded positively and confirmed that they have some varying knowledge levels of OSS. Again, most of the respondents (50) were IT professionals (computer programmers, analysts, systems administrators, and others) while the remainder (12) were just computer users.

Table 2. Rankings and mean degree of knowledge on 14 selected OSS categories

Software category	N	Min	Max	Mean	Std. deviation
Applied fields	62	1	4	2.24	0.843
Assistive technology	62	2	5	3.16	0.927
Data storage	62	3	5	4.16	0.632
Networking and Internet	62	2	5	4.16	0.751
Education	62	1	5	3.40	1.063
File managers	62	3	5	3.98	0.689
Games	62	2	5	4.18	0.758
Graphical user interface (GUI)	62	5	5	5.00	0.000
Health care	62	2	5	2.92	0.836
Media	62	2	5	4.45	0.761
Operating system	62	4	5	4.74	0.441
Password management	62	2	5	4.37	0.773
Programming language support	62	3	5	4.29	0.733
Security	62	4	5	4.90	0.298
Valid					

Knowledge for OSS is extremely important where adoption may follow. Bradbury (2006) notes that the availability and knowledge of programming languages such as Perl and PHP "have enabled us to create both ad hoc and full blown solutions in support of our infrastructure".

Table 2 shows that more knowledge has been placed on graphical user interface (GUI) systems (Mean = 5.00), Networking and Internet (Mean = 4.16), data storage (Mean = 4.16), games (Mean = 4.18), Media (Mean = 4.45), operating systems (Mean = 4.74), password management (Mean = 4.37), programming language (Mean = 4.29) and security systems (Mean = 4.90). This result is a clear indication that respondents had close to abundant knowledge of these systems. On the contrary, very little emphasis was put on applied fields (Mean = 2.24) and health care systems (Mean = 2.92). From these results, one can easily infer that most organizations are involved in software packages that they perceive can enhance and support their proprietary packages. The low scores in applied fields and health care systems indicate a low degree of knowledge for these rather complex systems. Other important findings, captured from part 2 (products known and/or being used), were that:

1. Data storage management systems have very good and reliable backup software, file archivers, search engines and database management systems (DBMS), among others.
2. Networking and Internet provided friendly email and instant messaging systems. Other components of this category included the webcam, web browsers, file transfer, file sharing across networks and several others.
3. Most respondents mentioned that some games that they play when bored included puzzle games, chess and speed racing, among others.
4. In media, several packages are known with most respondents particularly mentioning graphics, radio, television, video players, CD-writing software and audio editing management tools as the most commonly used.
5. Most respondents who claimed to have a lot of knowledge of operating and password management systems mentioning the most common of these systems as Linux and Open Solaris (operating systems) and Keepass and Password Safe (password management systems).
6. On programming language support systems, IT professionals, particularly computer programmers, analysts and others cited that they spent most of their time downloading bug trackers, code generators and version control systems, among others, for testing and eventual use within their infrastructures.
7. Security had a major impact on mean knowledge of software with all the 62 (100%) respondents indicating that they possessed abundant knowledge of anti-virus and other security systems. Most respondents mentioned that the most common antivirus software they download and use are Avira, Avast, AVG and Kaspersky.

The above findings can also be supported by the magnitudes of the standard deviation (Std. dev.) for each software class/category, Table 2. It must be noted that, the lower is the standard deviation for a category, the higher is the level of knowledge about it by the respondents. For instance, the standard deviations for GUI (Std. dev. = 0.000), operating systems (Std. dev. = 0.441) and security (Std. dev. = 0.298) reveal that most of the respondents in the sample have a clearer knowledge of these systems than they have on education (Std. dev. = 1.063), Assistive technology (Std. dev. = 0.927), applied fields (Std. dev. = 0.843) and health care (Std. dev. = 0.836) systems. The respondents spend more time on searching for, downloading, studying and eventually using systems such as GUI, OS and security than they do in education, assistive technology, applied fields and health care.

Table 3. Percentage distribution by software category

Software category		None	Very little	Little	Much	Abundant	Total
Applied fields	Frequency	13	24	22	3	0	62
	%	21.0	38.7	35.5	4.8	0.0	100
Assistive technology	Frequency	0	17	23	17	5	62
	%	0.0	27.4	37.1	27.4	8.1	100
Data storage	Frequency	0	0	8	36	18	62
	%	0.0	0.0	12.9	58.1	29.0	100
Education	Frequency	4	7	19	24	8	62
	%	6.5	11.3	30.6	38.7	12.9	100
File manager	Frequency	0	0	15	33	14	62
	%	0.0	0.0	24.2	53.2	22.6	100
Games	Frequency	0	2	7	31	22	62
	%	0.0	3.2	11.3	50.0	35.5	100
GUI	Frequency	0	0	0	0	62	62
	%	0.0	0.0	0.0	0.0	100.0	100
Healthcare	Frequency	0	22	25	13	2	62
	%	0.0	35.5	40.3	21.0	3.2	100
Media	Frequency	0	2	4	20	36	62
	%	0.0	3.2	6.5	32.3	58.1	100
OS	Frequency	0	0	0	16	46	62
	%	0.0	0.0	0.0	25.8	74.2	100
Password protection	Frequency	0	2	5	23	32	62
	%	0.0	3.2	8.1	37.1	51.6	100
Programming language support	Frequency	0	0	10	24	28	62
	%	0.0	0.0	16.1	38.7	45.2	100
Security	Frequency	0	0	0	6	56	62
	%	0.0	0.0	0.0	9.7	90.3	100
**		17	76	138	246	329	

Notes: ** The "no knowledge" hits were the lowest at 17, followed by "very little knowledge" (76), "little" (138), "much" (246) and "abundant" (329). These hits show that, generally, the respondents have some good knowledge of OSS systems.

It is also important to show the impact of each software category in terms of how respondents rated them. Table 3 illustrates how all the respondents faired per software category. With the exception of applied fields and education, all respondents show some knowledge (ranging from "very little", through "little", "much" to "abundant") for all the other twelve categories. All 62 (100%) respondents said they had very much knowledge of GUI systems. The other categories with high percentage rates of "abundant knowledge" were OS (74.2%), security (90.3%) followed by media (58.1%), password protection (51.6%) and programming language support (45.2%). The lowest recordings were in applied fields (0.0%), assistive technology (8.1%) and health care (3.2%).

These knowledge levels on all the products can be boosted by adopting what other countries have done. For example, Waters (2007) explores the benefits derived by U.S. schools from OSS. He notes that a growing number of schools are converting to OSS to cope with tight budgets and to get rid of vendor lock-in often associated with proprietary software.

Asked to comment on whether OSS could be an alternative for proprietary software, a university respondent quipped:

"*In this University, you have no choice but to use Windows and your preference doesn't matter. There isn't any initiative and commitment to OSS. As an IT professional, OSS is definitely a great idea because it costs nothing and sharpens your skills. It also gives you the flexibility to customize it the way you want. Windows is LOCKED!!!*"

On another note which shows that in some parts of the world OSS is actually known, a manager at The Ohio State University has just written in our daily mailing list:

"*We have an ancient "appliance" firewall here running OpenBSD 2.8 and IPFilter v3.3.18. I'm planning on upgrading the hardware, and this will lead to an OS change/upgrade. It doesn't look like IPFilter is supported in versions of OpenBSD later than 3.5, so I'm looking at either FreeBSD or NetBSD. (I need to be able to use IPFilter in the most recent stable release.) Are there any reasons for favoring one BSD variant over another in a firewall application?*"

This manager ends up with a note which says, "** E-mail is the best way to contact me **".

Discussion and conclusions

The findings of this study provide some very important and useful insight into OSS initiatives in Botswana in particular and Africa in general. Questions being asked as to whether the programmers in Africa know how to use OSS, whether programmers trust Windows to OSS and why OSS is finding it hard gaining traction in Africa, may well have been given some good ground through the respondents' answers, as well as a platform on which to continue being pursued. In conclusion, results have shown that the general feeling in Botswana is that, OSS is well known, often used, reliable, secure, easy to download, preferred, trusted and a candidate for adoption. Whether this knowledge is just a matrix of perceptions or facts will be known in further research initiatives.

Implications for research

This paper examined the question: "Does Botswana have enough knowledge about open source software?"

We discovered that the knowledge of various OSS systems is so much although it is now left to organizations as to whether they can adopt OSS or not. Almost all respondents indicated that they use OSS in one way or the other in their day-to-day company activities. Although the respondents, mainly programmers, analysts, systems administrators and web designers, expressed their desire to adopt OSS, claiming abundant knowledge of the same, it was their managers who hung onto proprietary software because according to them, "it is paid for", "it is more stable" (however with no comparison), "it is licensed and authentic" and "it is a company policy not to adopt OSS". Various other reasons including fear to change were advanced by the respondents, speaking on behalf of their managers. As a result, although there is so much knowledge about OSS, it is the desire of the organization to maintain the status quo of running the heavily expensive and sometimes unreliable proprietary systems. The adoption of OSS, on the other hand, may lead to organizations realizing higher returns as companies will not pay heavily towards the acquisition and implementation. Their focus would shift to improving the free systems only. The findings of this paper further suggest that the knowledge which IT professionals have acquired can be tamed to create a vibrant OSS platform in Botswana, resulting in huge Government savings on IT. This knowledge can be further strengthened by creating a culture of communication through email and mailing lists as identified by Sowe et al. (2005). They discuss how to construct and use an affiliation network. The mailing list, they argue, enables knowledge seekers and knowledge providers to interact to gain more knowledge. Lee (2006) has noted that governments throughout the world are now adopting different legislative and administrative strategies that support OSS while others direct public funds to large-scale OSS projects.

Perhaps it is important to note that, Iran depends, to a large extent, on OSS because of the U.S. embargo. It is also encouraging to note that almost half of all software engineers are women, Vaisam (2007). Given this vast knowledge of OSS products, future research must focus on OSS readiness – is Botswana ready to join the OSS initiatives?

The National Information and Communications (ICT) Policy in Botswana spells out an ambitious and exciting project that includes, among others, Connecting Communities Programme (CCP), the Government-online and the e-Health. It is clearly envisaged that these projects will disseminate huge volumes of information requiring the invoking of extra computing power in the form of OSS, available freely and cheaply. Thus, the ICT department of the Ministry of Transport and Communications should put in place a task force that will:

1. Capitalize on this OSS knowledge advantage with a view to establish the readiness of the ICT sector in Botswana to use OSS products.
2. Mobilize efforts aimed at growing, developing and sustaining a free OSS movement through mailing lists and other channels, resulting in the creation of a vibrant OSS community or programmers, analysts and users, among others.
3. Build knowledge and commitment to free OSS through communication, again using mailing lists and other tools.
4. Advocate for change management.
5. Create a conducive environment for the purpose of rolling out ICT towards development.
6. Encourage Botswana IT professionals and companies to accept and use free OSS. This will ensure that the region and Africa will also benefit from this initiative.
7. Support skill building schemes in the employment as well as implementation of free OSS systems.

It is important to note that OSS has had a hard time gaining traction in most of the African continent. This may probably be due to the fact that: (1) programmers in Africa regard Windows products as more superior to OSS; (2) programmers in Africa do not understand open source software products; and (3) that there is a general resistance to change by the providers of information technology budgets. Literature on OSS is loaded with examples from other parts of the world minus Africa, thereby creating a huge gap. A study of OSS initiatives in Africa would fill this yawning gap. More importantly, governments in Africa must pronounce public policy decisions on open source software – a step that will largely promote a serious consideration of OSS implementation.

If all this is done, the 2016 vision which, among others, envisages that all Batswana should be connected, can be achieved with OSS being used as a vehicle to take the nation of Botswana to this vision.

References

1. Agerfalk, P.J. and Fitzgerald, B. (2008). "Outsourcing to an Unknown Workforce: Exploring Open Sourcing as a Global Sourcing Strategy", MIS Quarterly, Vol. 3, No. 2, June 2008, pp. 385-409.
2. Alexy, O. and Henkel, J. (2009). "Promoting the Penguin: Intraorganizational Implications of Open Innovation (February 17, 2009), Available at SSRN: http://ssrn.com/abstract=988363
3. Alexy, O. and Leitner, M. (2009). "Norms, Rewards, and their Effect on Motivation: the Case of Open Source Software Development", (August 13, 2009). Available at SSRN: http://ssrn.com/abstract=1007689
4. Alkhatib, J., Anis, M. and Noori, H. (2008). "Open Source: the next big thing in technology transfer to developing nations", *International Association for Management of Technology*, IAMOT 2008 Proceedings.
5. Bagozzi, R.P. and Dholakia, U.M. (2006). "Open Source Software: Private Provision of a Public Good", Management Science, Vol. 52, No. 7, July 2006, pp. 1099-1115, DOI: 10.1287/mnsc.1060.0545.
6. Bessen, J. (2004). "Open Source Software: Free Provision of Complex Public Goods", 8-04.
7. Bessen, J. (2004). "What Is Free Software?"
8. Bradbury, D. (2006). "SMEs and Open Source – a perfect marriage".
9. Coar, K. (2007), "Open Source Initiative", *The Open Source Definition*, Version 19, http://www.opensource.org/docs/definition.php, 2007.
10. Comino, S. and Fabio, M.M. (2005). "Government Policies Supporting Open Source Software for the Mass Market," Review of Industrial Organization 26, No. 2 (March 2005), pp. 217-240.
11. Cox, M., Runge, C. and van de Ven, A. (2005). "Security Features in Red Hat Enterprise Linux 4", Revision 4b, February 2005.
12. Del Amo, M.Q. (2007). "Open Source Software: critical review of scientific literature and other sources.
13. Demaziere, D., Horn, F. and Zune, M (2007). "The functioning of a Free Software Community: Entanglement of Three Regulation Modes – Control, Autonomous and Distributed", *Science Studies*, Vol. 20 (2007), No. 2, pp. 34-54.
14. B. Fitzerald (2004). A critical look at open source, IT Systems Perspectives, 10, pp. 92-94.
15. Freeman, S. (2007). "The Material and Social Dynamics of Motivation: Contributions to Open Source Language Technology Development", Science Studies, Vol. 20, No. 2, pp. 55-77.
16. Guest Editorial, Free/Libre Open Source Software (FLOSS) (2007). Science Studies, Vol. 20, No. 2, pp. 5-6.
17. Henkel, J. (2008). "Champions of Revealing – the Role of Open Source Developers in Commercial Firms", CEPR, London.
18. Henkel, J. and Kas, S. (2008). "The emergence of openness: how firms discover free revealing as a means to appropriate value", *Working paper*.
19. Johnson, J.P. (2006). "Open Source Software: Private Provision of a Public Good", *Management Science*, Vol. 52.
20. G. Krogh, S. Spaeth and K. Lakhani (2003). "Community, joining, and specialization in open source software innovation: a case study", *Research Policy* 32, pp. 1217-1241.
21. Krishnamurthy, S. (2006). "On the Intrinsic and Extrinsic Motivation of Free/Libre/Open Source (FLOSS) Developers", Knowledge, Technology & Policy 18, No. 4 (Winter), pp. 17-39.
22. Larsen, M.H., Holck, J. and Pedersen, M.K. (2000). "The Challenges of Open Source Software in IT Adoption: enterprise Architecture versus Total Cost of Ownership", IRIS27.
23. Lattermann, C. and Stieglitz, S. (2005). "Framework for Governance in Open Source Communities", 2005 IEEE, Proceedings of the 38th Hawaii International Conference on System Sciences – 2005.
24. Lee, J (2006). "Government Policy toward Open Source Software: the Puzzles of Neutrality and Competition", Knowledge, Technology & Policy 18, No. 4 (Winter), pp. 113-141.
25. J. Lerner and J. Tirole (2001). The open source movement: key research questions, *European Economic Review*, 45 (5), pp. 819-826.
26. McGhee, D. (2007). "Free and Open Source Licenses: Benefits, Risks and Steps toward Ensuring Compliance", *Intellectual Property & Technology Law Journal*, 19, No. 11 (November), pp. 5-10.
27. McGowan, M.K., Paul, S. and Dexter, G. (2007). "An Exploration of the ideologies of Software Intellectual Property: the Impact on Ethical Decision Making", *Journal of Business Ethics*, 71, No. 4 (July 2007): 409-424.
28. Millar, C.C., Choi, C.J., Russell, E.T. and Kim, J. (2005). "Open source communities: an integrally informed approach", *Journal of Organizational Change Management*, Vol. 18, No. 3, Emerald Group Publishing Limited ISSN 0953-4814, pp. 259-268.
29. Moon, N.W. and Baker, P.M.A. "Adoption of Open Source Software: Preliminary Literature Review".
30. Muir, S.P. (2005), "An introduction to the open source software issue", Library Hi Tech, Vol. 23, No. 4, pp. 465-468, Emerald Group Publishing Limited, 0737-8831.
31. Van de Nieuwenhof, S. (2008). "Licensing Freedom: an Ethical Analysis of Free and Open Source Software Licenses".
32. Riehle, D. (2007). "The Economic Motivation of Open Source Software: Stakeholder Perspectives", IEEE Software, Published by IEEE Computer Society, 2007.
33. Sowe, S.K., Angelis, L., Stamelos, I. and Manolopoulos, Y. (2007). "Using Repository of Repositories (RoRs) to Study the Growth of F/OSS Projects: a Meta-Analysis Research Approach".
34. Tere, V. (2006). "'Intellectual Property' and Knowledge Creation in Disorganisations", E-Learning, Vol. 3, No. 3, 2006.
35. Tong, T.W. (2004). "Free/Open Source Software: Education", Asia-Pacific Development Information Programme, e-Primers on Free/Open Source Software, UNDP.
36. Vaisam, D. (2007). "Coding a Revolution", *Foreign Policy*, 159 (March/April), p. 93.
37. Ven, K., Verelst, J. and Mannaert, H. (2008). "Should You Adopt Open Source Software?" IEEE Software, Published by IEEE Computer Society.